MULTIMEDIA UNIVERSITY®

STUDENT ID NO

□□□□□□□□□□

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 2, 2018/2019

HPB3021 –PARALLEL COMPUTING

(All sections / Groups)

11 March 2019
9:00 -11:00 AM
(2 hours)

INSTRUCTIONS TO STUDENTS

1. This question paper consists of 5 pages, including this cover page.
2. You are required to attempt all questions. All questions carry equal marks (10).
3. Write all your answers in the Answer Booklet provided.
4. You may use a calculator.

Question 1

a) What is the difference between *chmod* and *chown*? [1 mark]

b) A bioinformatician downloaded a source file for a comparative genomic application program. The rules are the program should be executable by all, read by owner and group and written by owner only. Write out the command for the stated rules. [2 marks]

c) All the programs and files are stored as file in the Linux. What object holds the identity of the files? Explain. [2 marks]

d) An annotation was conducted on a draft genome. What command should be applied to investigate the number of genes annotated in the FASTA formatted file? [2 marks]

e) A scientist connected to a Linux system remotely through *ssh* program (eg: Putty) in a shared computer. The scientist run BLAST on 34,000 protein sequences against NCBI refseq database. What is the alternative that can be use to ensure the program to be ongoing although the remote connection is off? List any two commands as the alternatives. [3 marks]

Question 2

a) List down and explain TWO types of interconnection media in parallel computers by which the processors communicate with each other. Draw a diagram to demonstrate the processors communications via these two types of media. [3 marks]

b) Differentiate between "Asymmetrical Multicomputers" and "Symmetrical Multicomputers". List at least TWO differences. [2 marks]

c) Draw hypercube networks with two, four, and eight nodes. Make sure you label the nodes. Is a hypercube network with *n* nodes a subgraph of a hypercube network with *2n* nodes? [5 marks]

Continued ....

Question 3

a) Differentiate the definitions of "Parallel Computers" and "Parallel Programming". List and explain TWO categories of "Parallel Computers". [2 marks]

b) A formal way to identify parallelism in an activity is to draw a data dependency graph. Explain what a "Data Dependence Graph" is. List and explain two types of parallelism it presents with a short explanation. [3 marks]

c) Imagine we have the following block of code that is meant to be run on parallel computers. Draw a data dependence graph to perform the parallelism on the given code. Identify different types of parallelism on the graph and justify what is the optimal number of computers to process the code.

```
- A = 2
- B = A + 1
- C = A + 2
- D = 0
- While (D <= 500)
```

[5 marks]

Question 4

a) List any TWO commands to edit text in the terminal. [1 mark]

b) Why is it not advisable to use all the computing resource (eg: cores, RAM capacity) for multi-threaded processes? How do you check the availability of resource? [2 marks]

c) What is *X11*? [1 mark]

d) List any FOUR Linux shells. [1 mark]

e) Explain the definition of a task and a channel in task/channel model and describe the task/channel model. [2 marks]

f) The following C program is a MAIN function to parallelize a summation process using MPICH2 that calculates the sum of the elements of an array. The root process acts as a master and broadcasts the array to all child processes. Complete the blanks according to the given comments in the code. [3 marks]

Continued ....

```
/* mph header file
i.  <INCLUDE YOUR LINE HERE>
*/
# include <stdio.h>

int main ( int argc, char *argv[] ){

# define N 100

  double array[N];  int i;  int master = 0;  int my_id;
  int numprocs;  double PI = 3.141592653589793238462643;
  double seed;  MPI_Status status;  double sum;
  double sum_all;

/*Initialize MPI.
ii. <INCLUDE YOUR LINE HERE>
/*
  Get the number of processes.
*/
iii. <INCLUDE YOUR LINE HERE>
/*
  Determine the rank of this process.
*/
iv. <INCLUDE YOUR LINE HERE>
/*
  The master process initializes the array.
*/
  if ( my_id == master )
  {
    seed = 1.2345;

    for ( i = 0; i < N; i++ )
    {
      array[i] = ( double ) i * seed * PI;
    }
  }
/*
  The master process broadcasts the computed initial
values
  to all the other processes.
*/
v. <INCLUDE YOUR LINE HERE>
/*

 /*
  Each worker process sends its sum back to the master
process.
*/
  if ( my_id != master )
```

```
   {
     MPI_Send ( &sum, 1, MPI_DOUBLE, master, 1,
MPI_COMM_WORLD );
   }
   else
   {
     sum_all = sum;
     for ( i = 1; i < numprocs; i++ )
     {
       MPI_Recv ( &sum, 1, MPI_DOUBLE, MPI_ANY_SOURCE, 1,
         MPI_COMM_WORLD, &status );

       sum_all = sum_all + sum;}   }

/*
  Terminate MPI.
*/
vi. <INCLUDE YOUR LINE HERE>
/*
  Terminate.
*/
   return 0;

# undef N
}
```

END OF PAPER